WORLDTS: WORLD MODELING FOR MULTIMODAL COVARIATE-AWARE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series forecasting is typically framed as learning a direct mapping from historical to future observations in the observation space. However, sequences of observations generally provide only a partial view of the dynamics of the underlying system, with future observations being shaped by latent dynamics. Recent latent-space forecasting methods thus achieve improved performance by predicting future observations from latent-space representations of historical observations rather than directly forecasting future observations in the observation space. Next, while future observations are also shaped by external factors, how to incorporate external, often multimodal, information into forecasting, so that it can shape latent-state formation and evolution directly, remains underexplored. We propose WorldTS, a world-modeling based forecasting framework that integrates multimodal covariates directly into the forecasting to further improve forecasting performance. Specifically, WorldTS employs a two-stage training strategy. First, it learns forecasting-relevant latent state dynamics conditioned on multimodal covariates, yielding encoded future states. Next, the learned state dynamics are frozen, and an observation decoder is trained to map the predicted future states back to future observations. Extensive experiments on 21 real-world datasets offer insight into WorldTS and its effectiveness.

1 INTRODUCTION

Time series forecasting plays a fundamental role in a wide range of applications, including energy, transportation, climate, finance, and public governance (Martínez-Álvarez et al., 2011; Qiu et al., 2024; Xu et al., 2026; Wu et al., 2024; Zheng et al., 2015). Traditional time series forecasting learns a direct mapping from historical observations to future observations in the observation space. However, observed observations are only external manifestations of the underlying states of dynamic systems, rather than the states themselves (Ghugare et al., 2023; Kaiser et al., 2020). For many systems, identical or similar observations may correspond to different underlying states, while future evolution largely depends on these states and their dynamics. Directly forecasting future observations from historical observations therefore entangles state evolution with its manifestation in the observation space. As a result, models may focus more on fitting numerical relationships in the observation space than on explicitly modeling the underlying system states and their dynamics.

This reasoning motivates a more natural alternative: explicitly modeling future states in a latent space, shifting time series forecasting towards modeling system states and their dynamics. Specifically, a model first extracts state representations from historical observations, predicts their future evolution in the latent space, and then decodes the predicted states back into future observations. Compared with directly forecasting future observations, this formulation treats future states as prediction targets, placing emphasis on latent-state evolution and partially decoupling state dynamics from observation generation. Modeling system evolution through future-state prediction in a latent space has become an important paradigm in world modeling (Assran et al., 2023; Maes et al., 2026). Similarly, in time series forecasting, mapping historical observations into a latent state space and modeling future states in that space can better capture system states and dynamics, thereby improving forecasting performance (Yang et al., 2026).

However, in real-world time series forecasting future states are rarely determined by historical states alone; they are also shaped by external factors. Historical observations describe past evolution,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

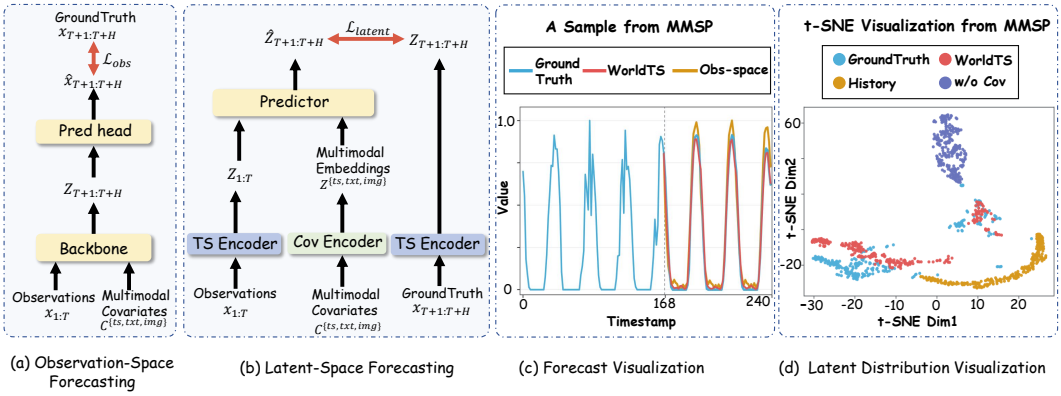

Figure 1: Comparison of observation-space and latent-space forecasting. (a) Observation-space forecasting supervised by ground-truth future observations. (b) Latent-space forecasting supervised by states encoded from the ground-truth future observations. (c) Forecasts on the Multimodal Solar Power (MMSP) dataset by WorldTS and its observation-space variant (Obs-space). (d) Latent distributions on MMSP, comparing predicted future states with states encoded from the ground-truth future observations.

whereas external factors in the form of covariates provide additional predictive information relevant to future evolution. For example, electricity-load dynamics depend not only on historical consumption patterns but also on external weather conditions, while photovoltaic power generation is affected by cloud cover (Ma et al., 2024; Wang et al., 2026). Forecasting future states solely from the historical observations may overlook important factors governing system evolution, and incorporating evolution-relevant covariates can provide richer information for modeling future system states.

Covariates often span multiple modalities. Different modalities can provide complementary information that is difficult to infer from historical observations, thus offering additional cues for predicting future system evolution (Liu et al., 2024a; Ma et al., 2024). Latent-space modeling provides a natural way to integrate multimodal information: diverse modalities can be encoded into compact representations that can then contribute to future-state prediction (Han et al., 2026; Li et al., 2026b). Consequently, multimodal covariates can enrich state modeling beyond historical data, and their latent representations can be incorporated naturally into future-state prediction.

Motivated by these observations, we propose **WorldTS**, a world-modeling framework for multimodal, covariate-aware time series forecasting. As illustrated in Figure 1(a), traditional covariate-aware forecasting predicts future observations directly and is optimized in the observation space. WorldTS instead performs prediction in a latent state space. As shown in Figure 1(b), in Stage 1, the shared TS Encoder maps both the historical observations and the ground-truth future observations into the latent state space, while modality-specific encoders (Cov Encoder) map multimodal covariates into their corresponding representations. The historical states and covariate representations are then used to predict future states, which are directly supervised by the states encoded from the ground-truth future observations. In Stage 2, we freeze the encoders and predictor and train a decoder to map the predicted future states back to the observation space. Figures 1(c)–(d) further illustrate how this design affects both the predicted states and the final forecasts. Figure 1(d) compares the latent distributions of the predicted future states and the states encoded from the ground-truth future observations. With covariates, the predicted future states exhibit a distribution more consistent with that of the encoded future states, whereas removing covariates results in a more pronounced distribution discrepancy. Figure 1(c) shows the corresponding forecasting results in the observation space. Here, Obs-space denotes a one-stage variant of WorldTS trained end-to-end using only an observation-space loss, without direct supervision on future states. Compared with Obs-space, WorldTS produces a forecast trajectory that more closely matches the ground-truth future sequence.

In summary, our main contributions are as follows:

- We formulate multimodal covariate-aware time series forecasting from a latent-space future-state prediction perspective, where multimodal covariates condition future target-state prediction rather than being used only for observation-space forecasting.
- We propose **WorldTS**, a world-modeling framework featuring direct supervision from states encoded from future observations and a two-stage training strategy. WorldTS first learns covariate-conditioned latent state dynamics and then decodes predicted states back into future observations.
- We report on extensive experiments on 21 real-world datasets, covering numerical-, text-, and image-covariate forecasting settings. Comprehensive comparisons and ablation studies characterize the effectiveness of WorldTS. Code and data are available at `https://anonymous.4open.science/r/WorldTS-C25A/`.

2 Related Work

2.1 Latent-Space Predictive Modeling

Latent-space prediction is central to world modeling, where observations are encoded as latent states whose future evolution is predicted. DreamerV3 (Hafner et al., 2023) learns latent state transitions from observation and action histories, supporting decision-making through imagined state trajectories. JEPA-based methods use encoded observations as explicit prediction targets. V-JEPA 2-AC (Assran et al., 2025) uses a pretrained video encoder to obtain state representations and predicts future frame representations from historical states and actions. LeWorldModel (Maes et al., 2026) learns latent states and their transitions directly from images, using encoded future observations to supervise action-conditioned state prediction.

Latent-space predictive modeling has also been explored for time series forecasting through representation alignment and future-state prediction. TimeAlign (Hu et al., 2026) aligns representations learned from historical inputs with those encoded from ground-truth future observations to guide representation learning. LatentTSF (Yang et al., 2026) maps historical and future sequences into a shared latent space, predicts future states within this space, and decodes them back into observations. These methods primarily encode and predict latent states from the target series itself. In contrast, WorldTS further incorporates multimodal covariates into future-state prediction, enabling the use of additional information beyond historical observations.

2.2 Covariate-aware Forecasting

Covariate-aware forecasting leverages external information to improve prediction. For numerical covariates, existing methods focus mainly on modeling interactions between endogenous and exogenous variables or on exploiting known future information. TimeXer (Wang et al., 2024) and CrossLinear (Zhou et al., 2025) capture dependencies between target and exogenous series, while TFT (Lim et al., 2021) and TiDE (Das et al., 2023) explicitly accommodate known future covariates. DAG (Qiu et al., 2026) further models cross-variable dependencies along both temporal and channel dimensions, GCGNet (Li et al., 2026a) introduces graph-structured consistency, and KITE (Cheng et al., 2026) incorporates external information into probabilistic forecasting.

Recent studies extend covariate-aware forecasting beyond numerical variables to cover heterogeneous modalities. TaTS (Li et al., 2026b) and VoT (Wang et al., 2026) exploit textual information through paired-text modeling and event-driven reasoning, respectively, while FusionSF (Ma et al., 2024) incorporates heterogeneous sources such as satellite imagery. UniCA (Han et al., 2026) provides a unified approach for adapting time-series foundation models to heterogeneous covariates. While these methods differ considerably, they primarily incorporate covariates as additional information for observation-space forecasting. In contrast, WorldTS considers latent-space future-state prediction: multimodal covariates are encoded into latent representations and are used to condition future-state prediction, while states encoded from ground-truth future observations provide direct supervision for the predicted states. This latent-space formulation provides a natural way to integrate heterogeneous multimodal information into future-state modeling.

3 PRELIMINARIES

3.1 PROBLEM FORMULATION

Multimodal Covariate-aware Forecasting. Given a historical time series $x_{1:T} \in \mathbb{R}^{T\times C}$ with C variables, the goal is to forecast $x_{T+1:T+H} \in \mathbb{R}^{H\times C}$ over the next H time steps using multimodal covariates:

$$\mathcal{C} = \left\{c^m_{1:\tau_m}\right\}_{m\in\mathcal{M}}, \qquad \mathcal{M} = \{\text{ts}, \text{txt}, \text{img}\}, \tag{1}$$

where τ_m denotes the temporal availability of modality m.[1] The forecasting task is formulated as

$$\widehat{x}_{T+1:T+H} = F_\theta(x_{1:T}, \mathcal{C}), \tag{2}$$

where F_θ denotes a forecasting model parameterized by θ.

Latent-State Forecasting. WorldTS further formulates the above task as forecasting future states in a learned latent space. The historical observations and the covariates from each modality are first mapped to their corresponding representations:

$$Z_{1:T} = E_\theta(x_{1:T}) \in \mathbb{R}^{T\times D}, \quad Z^m_{1:\tau_m} = E^m_{\theta_m}\left(c^m_{1:\tau_m}\right), \quad m \in \mathcal{M} \tag{3}$$

Here, E_θ is the state encoder, $E^m_{\theta_m}$ is the encoder for modality m, and D is the latent-state dimensionality. Conditioned on the historical states and encoded multimodal covariates, the predictor P_ψ predicts future states, which are subsequently mapped back to the observation space by the decoder D_ϕ:

$$\widehat{Z}_{T+1:T+H} = P_\psi\left(Z_{1:T}, \left\{Z^m_{1:\tau_m}\right\}_{m\in\mathcal{M}}\right), \qquad \widehat{x}_{T+1:T+H} = D_\phi\left(\widehat{Z}_{T+1:T+H}\right) \tag{4}$$

During training, the same state encoder encodes the ground-truth future observations to obtain

$$Z_{T+1:T+H} = E_\theta(x_{T+1:T+H}) \in \mathbb{R}^{H\times D} \tag{5}$$

which provides direct supervision for $\widehat{Z}_{T+1:T+H}$.

4 WORLDTS

4.1 FRAMEWORK OVERVIEW

As shown in Figure 2, WorldTS first learns state dynamics conditioned on multimodal covariates and then decodes the future states into observations. Specifically, its first stage, Latent Dynamics Learning, learns latent dynamics conditioned on multimodal covariates. Historical observations and multimodal covariates are encoded through separate branches into historical states and covariate representations. The predictor combines these representations to predict future states. Meanwhile, the shared TS encoder encodes the ground-truth future observations to directly supervise state prediction. State-matching losses and anti-collapse regularization jointly optimize the learnable encoding modules and predictor. In its second stage, Observation Decoding, WorldTS learns to decode future states into observations. The encoders and predictor are frozen, and only the decoder is trained. Predicted future states and states encoded from the ground-truth future observations pass through the same decoder. Both outputs are supervised by ground-truth future observations, yielding prediction and reconstruction losses, respectively. During inference, WorldTS predicts future states using only the historical observations and available covariates, then decodes them into future observations.

We next elaborate on the key modules introduced above, including the TS Encoder and TS Decoder (Section 4.2), multimodal covariate encoders (Section 4.3), and covariate-conditioned state predictor (Predictor, Section 4.4), followed by the two-stage training objectives (Section 4.5).

[1] Covariates may be future-unknown ($\tau_m = T$) or future-known ($\tau_m = T + H$).

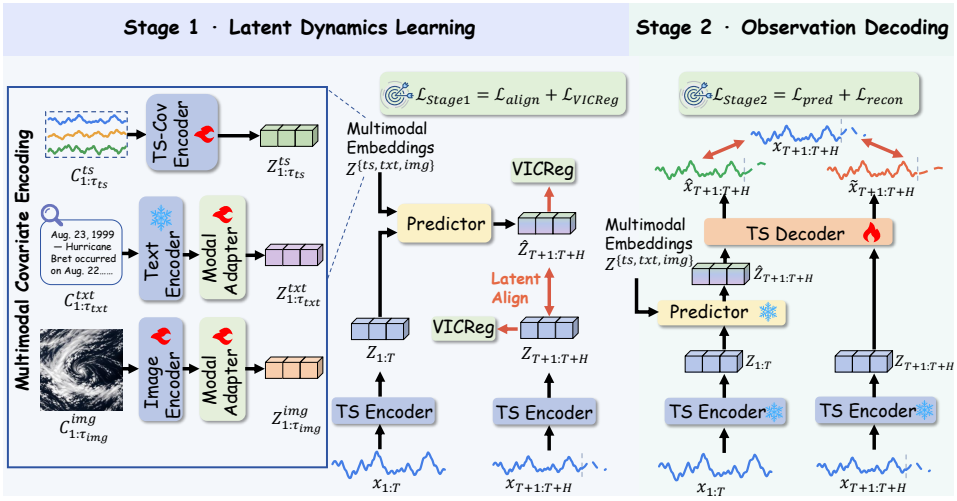

Figure 2: Overview of WorldTS. TS Encoder/Decoder denote the time-series encoder/decoder; Predictor denotes the covariate-conditioned state predictor. TS-Cov Encoder, Text/Image Encoder, and Modal Adapter denote the numerical-covariate encoder, modality-specific feature encoders, and MLP projector, respectively. Stage 1 (Latent Dynamics Learning) learns future-state prediction conditioned on multimodal covariates. Stage 2 (Observation Decoding) freezes the encoders and predictor and trains the decoder to recover future observations.

4.2 TS Encoder and TS Decoder

In Stage 1, the shared TS Encoder encodes the historical observations and the ground-truth future observations into a common latent space. In Stage 2, the TS Decoder learns to map future states back to observations. Both use local patches to capture temporal context (Ekambaram et al., 2023); their architectures are illustrated in Figure 5 in the appendix.

We apply $\mathrm{CausalPatching}_{S_e}$ to the observations $x_{1:L} \in \mathbb{R}^{L\times C}$, extracting L overlapping length-S_e patches with stride one and replicate padding at the beginning. The patch ending at time t contains only observations up to t, thereby preserving causality. Each patch is flattened and mapped through a shared linear projection into the D-dimensional state space:

$$U^x = \mathrm{Linear}_e\big(\mathrm{CausalPatching}_{S_e}(x_{1:L})\big) \in \mathbb{R}^{L\times D} \tag{6}$$

A residual two-layer feed-forward network with GELU activation then refines the projected patches:

$$Z_{1:L} = E_\theta(x_{1:L}) = U^x + \mathrm{FFN}_e(U^x) \in \mathbb{R}^{L\times D} \tag{7}$$

The same TS Encoder maps the historical observations and the ground-truth future observations into a shared state space, with the last $S_e - 1$ historical observations providing context for the first future patches. This produces the historical states $Z_{1:T}$ and future states $Z_{T+1:T+H}$, which supervise the predicted states $\widehat{Z}_{T+1:T+H}$ during training.

For decoding with patch length S_d, we concatenate the last $S_d - 1$ historical states with the predicted future states:

$$\overline{Z} = \left[Z_{T-S_d+2:T}; \widehat{Z}_{T+1:T+H}\right] \in \mathbb{R}^{(S_d-1+H)\times D} \tag{8}$$

The decoder extracts H consecutive length-S_d state patches from $\overline{Z}$, one per forecast step, and projects each patch into a D-dimensional decoded representation:

$$U^z = \mathrm{Linear}_d\big(\mathrm{Patching}_{S_d}(\overline{Z})\big), \qquad U^z \in \mathbb{R}^{H\times D} \tag{9}$$

A residual two-layer feed-forward network with GELU activation refines these representations:

$$V^z = U^z + \mathrm{FFN}_d(U^z) \tag{10}$$

The output projection finally maps the refined representations from the state dimensionality D to the observation dimensionality C:

$$\widehat{x}_{T+1:T+H} = D_\phi\Big(\widehat{Z}_{T+1:T+H}\Big) = \mathrm{Linear}_o(V^z) \in \mathbb{R}^{H\times C} \tag{11}$$

4.3 Multimodal Covariate Encoding

In parallel with the TS Encoder, modality-specific encoders map covariates into representations for the Predictor. The numerical-covariate encoder (TS-Cov Encoder) follows the TS Encoder architecture with separate parameters:

$$\begin{aligned} U^{\mathrm{ts}} &= \mathrm{Linear}_{\mathrm{ts}}\big(\mathrm{CausalPatching}_{S_e}\big(c^{\mathrm{ts}}_{1:\tau_{\mathrm{ts}}}\big)\big) \\ Z^{\mathrm{ts}}_{1:\tau_{\mathrm{ts}}} &= E^{\mathrm{ts}}_{\theta_{\mathrm{ts}}}\big(c^{\mathrm{ts}}_{1:\tau_{\mathrm{ts}}}\big) = U^{\mathrm{ts}} + \mathrm{FFN}_{\mathrm{ts}}(U^{\mathrm{ts}}) \end{aligned} \tag{12}$$

The text branch follows VoT (Wang et al., 2026) and extracts textual features using the input embedding layer of a frozen GPT-2 model, whereas the image branch follows UniCA (Han et al., 2026) and employs a lightweight CNN to extract visual features. Each branch then maps the extracted features through a lightweight MLP projector (Modal Adapter) to obtain Z^{txt} or Z^{img}. Architectural and implementation details are provided in the appendix.

4.4 Covariate-Conditioned State Predictor

As shown in Figure 3, the Predictor P_ψ fuses outputs from the preceding encoders to predict base future states, then refines them with future covariates when available.

Specifically, the historical states $Z_{1:T}$ and encoded historical numerical covariates $Z^{\mathrm{ts}}_{1:T}$ are concatenated along the feature dimension and mapped into fused historical states $S^{\mathrm{num}}_{1:T} \in \mathbb{R}^{T\times D}$:

$$S^{\mathrm{num}}_{1:T} = \mathrm{MLP}_h\big(\mathrm{Concat}\big[Z_{1:T}; Z^{\mathrm{ts}}_{1:T}\big]\big) \tag{13}$$

where MLP_h uses layer normalization and GELU activation to produce the fused historical states $S^{\mathrm{num}}_{1:T}$. Without numerical covariates, we set $S^{\mathrm{num}}_{1:T} = Z_{1:T}$.

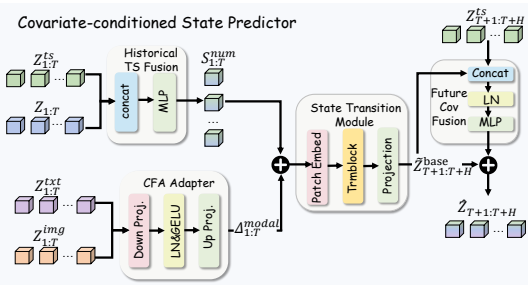

Figure 3: Covariate-conditioned state predictor. Historical covariates are integrated into the historical states through direct fusion or CFA, while future covariates further refine the predicted states.

The encoded text and image representations $Z^{\mathrm{modal}}_{1:T} \in \{Z^{\mathrm{txt}}_{1:T}, Z^{\mathrm{img}}_{1:T}\}$ are aligned with the historical timeline. To filter information irrelevant to the state dynamics, the Controlled Fusion Adapter (CFA) (Lee et al., 2026) uses a low-rank bottleneck to map these representations into residuals in the state space:

$$\Delta^{\mathrm{modal}}_{1:T} = \mathrm{Dropout}\big(\sigma\big(\mathrm{LN}\big(Z^{\mathrm{modal}}_{1:T} W_{\mathrm{down}}\big)\big)\big)\, W_{\mathrm{up}} \in \mathbb{R}^{T\times D}, \tag{14}$$

where W_{down} projects the modality representation into a low-dimensional bottleneck, W_{up} maps it to the D-dimensional state space, and σ denotes GELU activation. The resulting $\Delta^{\mathrm{modal}}_{1:T}$ is added to $S^{\mathrm{num}}_{1:T}$, introducing text or image information while retaining the historical state as the primary predictive signal.

The fused historical states are then processed by the state-transition module P^{trans}_ψ, which follows a patch-based Transformer design (Nie et al., 2023) to predict the base future-state trajectory:

$$\widetilde{Z}^{\mathrm{base}}_{T+1:T+H} = P^{\mathrm{trans}}_\psi\big(S^{\mathrm{num}}_{1:T} + \Delta^{\mathrm{modal}}_{1:T}\big) \in \mathbb{R}^{H\times D} \tag{15}$$

When future covariates are available, their representations are combined with the base future states to compute a state residual:

$$\Delta^{\mathrm{future}}_{T+1:T+H} = \mathrm{MLP}_f\Big(\mathrm{LN}\Big(\mathrm{Concat}\Big[\widetilde{Z}^{\mathrm{base}}_{T+1:T+H}; Z^{\mathrm{ts}}_{T+1:T+H}\Big]\Big)\Big) \in \mathbb{R}^{H\times D} \tag{16}$$

The final future states are:

$$\widehat{Z}_{T+1:T+H} = \widetilde{Z}^{\text{base}}_{T+1:T+H} + \Delta^{\text{future}}_{T+1:T+H} \tag{17}$$

Without future covariates, the base prediction is used directly. The resulting future-state predictions receive latent-space supervision in Stage 1 and are decoded into observations in Stage 2.

4.5 Two-Stage Learning Objectives

Stage 1: Latent Dynamics Learning. The predictor outputs future-state predictions $\widehat{Z}$, supervised by states Z obtained by encoding the ground-truth future observations with the shared TS encoder. Inspired by LatentTSF (Yang et al., 2026), we match these states using an element-wise L_1 loss and cosine similarity:

$$\mathcal{L}_{\text{latent}} = \left\| \widehat{Z} - \text{sg}(Z) \right\|_1, \qquad \mathcal{L}_{\text{sim}} = 1 - \cos\Big(\widehat{Z}, \text{sg}(Z)\Big) \tag{18}$$

Here, $\text{sg}(\cdot)$ stops gradients; cosine similarity is computed over flattened future state sequences.

To prevent representation collapse during state matching, we adopt the variance and covariance regularizers from VICReg (Bardes et al., 2022) for both predicted and encoded future states, with statistics computed across samples and forecast steps:

$$\begin{gathered} \mathcal{V}(A) = \frac{1}{D}\sum_{d=1}^{D}\left[1 - \sqrt{\text{Var}(A_{:,d}) + \epsilon}\right]_+, \qquad \mathcal{R}_{\text{cov}}(A) = \frac{1}{D}\sum_{i\neq j}[\text{Cov}(A)]_{ij}^2, \\ \mathcal{L}_{\text{VICReg}} = \frac{1}{2}\sum_{A\in\{Z,\widehat{Z}\}}\left[\mathcal{V}(A) + \lambda_{\text{cov}}\mathcal{R}_{\text{cov}}(A)\right] \end{gathered} \tag{19}$$

We jointly optimize the learnable encoding modules and predictor with the following objective:

$$\mathcal{L}_{\text{Stage1}} = \lambda_{\text{sim}}\mathcal{L}_{\text{sim}} + \lambda_{\text{latent}}\mathcal{L}_{\text{latent}} + \lambda_{\text{VIC}}\mathcal{L}_{\text{VICReg}} \tag{20}$$

Stage 2: Observation Decoding. With the encoders and predictor frozen, $\widehat{Z}$ and Z are passed through the same TS decoder. Both outputs are supervised by the ground-truth future observations x, yielding prediction and reconstruction losses that update only the decoder:

$$\mathcal{L}_{\text{Stage2}} = \underbrace{\left\| x - D_\phi(\widehat{Z}) \right\|_1}_{\mathcal{L}_{\text{pred}}} + \alpha \underbrace{\left\| x - D_\phi(Z) \right\|_1}_{\mathcal{L}_{\text{recon}}} \tag{21}$$

The prediction term adapts the decoder to states available at inference; the reconstruction term provides additional supervision from states encoded from the ground-truth future observations.

5 Experiments

5.1 Experimental Settings

Datasets. We evaluate WorldTS on 21 real-world datasets: 12 numerical-covariate datasets used by DAG (Qiu et al., 2026), eight text-covariate Time-MMD datasets (Liu et al., 2024a), and the image-covariate MMSP solar-power dataset (Ma et al., 2024). Dataset details are in Appendix A.1.

Baselines. For numerical-covariate forecasting, we compare WorldTS with ten baselines, covering dedicated covariate-aware methods and general forecasting backbones, including KITE (Cheng et al., 2026) and DUET (Qiu et al., 2025). For text-covariate forecasting, we compare with nine baselines covering time-series-only and multimodal methods, including PatchTST (Nie et al., 2023) and VoT (Wang et al., 2026). For image-covariate forecasting, following UniCA (Han et al., 2026), we compare with thirteen baselines spanning specialized forecasters, pretrained time-series models, multimodal methods, and UniCA-adapted models, including TTM-R2 (Ekambaram et al., 2024) and FusionSF (Ma et al., 2024). The full baseline list and descriptions are provided in the appendix.

Implementation Details. We implement WorldTS in PyTorch on NVIDIA A800 80GB GPUs and report MSE and MAE. For numerical-covariate, text-covariate, and image-covariate forecasting,

Table 1: Average results over short and long forecasting horizons on the 12 numerical-covariate datasets. Baseline results are from DAG (Qiu et al., 2026) and KITE (Cheng et al., 2026) under the same evaluation protocol. Full results: Table 5 (Appendix B). **Red**: best; blue: second best.

Models	**WorldTS** (ours)		KITE (2026)		DAG (2026)		GCGNet (2026)		DUET (2025)		CrossLinear (2025)		Amplifier (2025)		TimeKAN (2025)		TimeXer (2024)		TiDE (2023)		TFT (2021)	
Metrics	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae
NP	**0.297**	**0.319**	0.325	0.323	0.362	0.344	0.370	0.348	0.411	0.407	0.371	0.387	0.420	0.418	0.405	0.419	0.418	0.371	0.443	0.400	0.379	0.375
PJM	**0.087**	**0.178**	0.096	**0.178**	0.093	0.180	0.095	0.187	0.102	0.197	0.112	0.223	0.137	0.246	0.139	0.262	0.108	0.198	0.142	0.246	0.114	0.207
BE	**0.393**	**0.260**	0.427	0.286	0.423	0.279	0.431	0.294	0.514	0.354	0.479	0.337	0.559	0.413	0.548	0.407	0.452	0.290	0.498	0.325	0.454	0.291
FR	**0.384**	**0.210**	0.387	0.225	0.414	0.219	0.415	0.234	0.495	0.327	0.482	0.298	0.554	0.408	0.547	0.374	0.427	0.241	0.484	0.281	0.504	0.257
DE	**0.340**	**0.353**	0.350	0.370	0.370	0.370	0.401	0.389	0.482	0.430	0.485	0.452	0.473	0.441	0.473	0.445	0.475	0.418	0.499	0.447	0.489	0.446
Energy	**0.091**	**0.226**	0.111	0.257	0.124	0.267	0.131	0.277	0.203	0.367	0.239	0.401	0.233	0.389	0.218	0.381	0.163	0.315	0.153	0.302	0.130	0.283
Sdwpfm1	**0.391**	**0.412**	0.417	0.451	0.423	0.461	0.424	0.457	0.599	0.570	0.426	0.502	0.437	0.490	0.447	0.534	0.701	0.609	0.483	0.507	0.482	0.474
Sdwpfm2	**0.446**	**0.450**	0.475	0.483	0.477	0.485	0.475	0.486	0.514	0.490	0.533	0.573	0.491	0.512	0.497	0.564	0.803	0.653	0.486	0.516	0.476	0.488
Sdwpfh1	**0.376**	**0.431**	0.441	0.491	0.448	0.486	0.450	0.500	0.539	0.516	0.557	0.593	0.537	0.598	0.577	0.638	0.746	0.643	0.453	0.508	0.479	0.491
Sdwpfh2	**0.419**	**0.461**	0.500	0.526	0.523	0.530	0.520	0.536	0.647	0.566	0.538	0.574	0.521	0.581	0.647	0.672	0.891	0.719	0.599	0.583	0.566	0.521
Colbun	**0.088**	**0.146**	**0.088**	0.172	0.098	0.154	0.107	0.175	0.198	0.266	0.126	0.195	0.173	0.246	0.128	0.175	0.145	0.235	0.164	0.227	0.238	0.297
Rapel	**0.228**	**0.259**	0.244	0.294	0.230	0.305	0.306	0.307	0.269	0.326	0.252	0.313	0.257	0.321	0.249	0.311	0.344	0.362	0.320	0.351	0.305	0.333
1st Count	**12**	**12**	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 2: Forecasting results on eight Time-MMD datasets with a prediction horizon of 12. Baseline results are taken from VoT (Wang et al., 2026) under the same evaluation setting.

Models	**WorldTS** (ours)		VoT (2026)		TaTS (2026)		Time-VLM (2025)		CALF (2025)		GPT4MTS (2024)		GPT4TS (2023)		RaFT (2025)		iTransformer (2024)		PatchTST (2023)	
Metrics	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae
Agriculture	**0.285**	**0.338**	0.288	0.355	0.290	0.350	0.322	0.359	0.314	0.355	0.301	0.342	0.291	**0.338**	0.301	0.383	0.300	0.352	0.309	0.351
Climate	**1.072**	**0.838**	1.088	0.845	1.179	0.885	1.203	0.896	1.177	0.883	1.152	0.876	1.171	0.883	1.295	0.930	1.131	0.864	1.168	0.877
Economy	**0.214**	**0.363**	0.223	0.374	0.229	0.381	0.245	0.396	0.216	**0.363**	0.226	0.379	0.233	0.386	0.280	0.425	0.230	0.385	0.229	0.382
Energy	**0.090**	**0.201**	0.091	0.218	0.105	0.232	0.114	0.253	0.102	0.224	0.111	0.244	0.111	0.243	0.123	0.255	0.121	0.258	0.107	0.235
Health	**0.778**	**0.542**	0.898	0.596	0.899	0.612	1.198	0.727	0.964	0.609	0.985	0.658	0.854	0.629	1.297	0.769	1.261	0.746	1.006	0.650
Security	**74.325**	**3.979**	75.931	4.153	77.656	4.239	80.767	4.438	74.631	4.113	78.041	4.316	76.965	4.219	83.911	4.712	82.644	4.493	77.815	4.254
SocialGood	0.962	**0.421**	**0.925**	0.438	1.053	0.474	1.005	0.505	0.991	0.439	1.093	0.470	1.167	0.608	1.151	0.553	1.153	0.574	1.263	0.643
Traffic	0.185	**0.238**	**0.181**	0.239	0.189	0.242	0.222	0.322	0.193	0.243	0.218	0.268	0.211	0.260	0.300	0.387	0.205	0.250	0.188	0.239
1st Count	**6**	**8**	2	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0

Table 3: Forecasting results on the MMSP with a prediction horizon of 24. Baseline results are taken from UniCA (Han et al., 2026) under the same evaluation setting.

Category	Ours	Multimodal		Specialized Forecasters					Pretrained Models			UniCA-adapted		
Models	**WorldTS**	MM-TSF	FusionSF	TFT	D.AR	NB.S	TiDE	P.TST	TTM-R2	Moirai	PFN-TS	Chronos-Bolt	TimesFM	Time-MoE
MSE	**0.061**	0.090	0.478	0.067	0.097	0.107	0.297	0.662	0.095	0.206	0.633	0.090	0.098	0.653
MAE	**0.171**	0.200	0.566	0.177	0.216	0.219	0.438	0.682	0.263	0.378	0.711	0.193	0.229	0.765

we follow the experimental settings of DAG (Qiu et al., 2026), VoT (Wang et al., 2026), and UniCA (Han et al., 2026), respectively. Dataset-specific lookback windows and prediction horizons are detailed in Appendix A.3. "Drop Last" is disabled at test time for all three settings to retain the final incomplete batch (Qiu et al., 2024).

5.2 Main Findings

Numerical-Covariate Forecasting. As shown in Table 1, WorldTS achieves the lowest MSE and MAE across all 12 datasets. On Energy, it reduces MSE by 18.0% relative to the strongest baseline. These results show its effectiveness across diverse numerical-covariate forecasting tasks.

Text-Covariate Forecasting. Across the eight Time-MMD datasets, WorldTS ranks first in 14 of 16 dataset–metric comparisons against the time-series-only and multimodal baselines. The largest gains are on Health: 8.9% in MSE and 9.1% in MAE relative to the strongest baselines.

Image-Covariate Forecasting. Table 3 reports results on MMSP, where WorldTS achieves the lowest MSE and MAE. Compared with TFT, the strongest baseline on both metrics, WorldTS reduces MSE and MAE by approximately 9.0% and 3.4%, respectively. These results show the effectiveness of WorldTS in leveraging numerical and visual covariates.

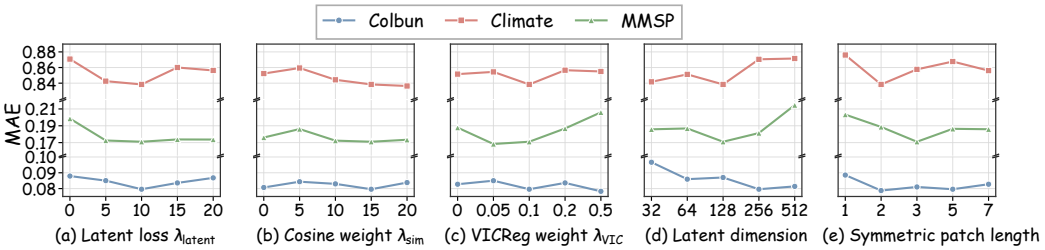

Figure 4: Parameter sensitivity of WorldTS. We vary the weights of three losses (λ_{latent}, λ_{sim}, and λ_{VIC}), the latent-state dimensionality, and the symmetric causal patch length one at a time on Colbun, Climate, and MMSP, and report MAE. Other settings remain fixed.

Table 4: Ablation results on representative datasets with numerical, textual, and visual covariates.

Dataset (MSE / MAE)	Full	w/o $\mathcal{L}_{\text{latent}}$	w/o $\mathcal{L}_{\text{sim}}$	w/o $\mathcal{L}_{\text{VICReg}}$	Pointwise MLP	One-stage MAE	w/o Covariates
Colbun	**0.052 / 0.080**	0.056 / 0.088	0.055 / 0.081	0.053 / 0.083	0.066 / 0.091	0.059 / 0.096	0.073 / 0.088
Climate	**1.072 / 0.838**	1.145 / 0.871	1.089 / 0.852	1.111 / 0.852	1.086 / 0.844	1.149 / 0.875	1.134 / 0.869
MMSP	**0.061 / 0.171**	0.078 / 0.199	0.063 / 0.176	0.068 / 0.188	0.065 / 0.187	0.085 / 0.217	0.101 / 0.279

5.3 Parameter Sensitivity

We assess parameter sensitivity on Colbun, Climate, and MMSP by varying one hyperparameter at a time. The results are shown in Figure 4. We make the following observations: 1) Setting any loss weight to zero increases MAE relative to its default on all three datasets, indicating that each loss contributes to prediction accuracy. 2) Among the tested values, $\lambda_{\text{latent}} = 10$ achieves the lowest MAE on all three datasets, while $\lambda_{\text{sim}} = 15$ and $\lambda_{\text{VIC}} = 0.1$ consistently achieve either the best or second-best performance. Increasing λ_{VIC} to 0.5 increases the MMSP error markedly, showing that stronger regularization does not necessarily improve performance. 3) Moderate latent-state dimensionalities perform better: 128 or 256 dimensions yield the lowest MAE, with no consistent gains from larger dimensions. 4) Moderate causal patch lengths generally perform better, suggesting that overly short or long temporal contexts can reduce forecasting accuracy.

5.4 Ablation Study

We compare the full model with six ablation variants and report the results in Table 4 and the protocols in Appendix A.4. We make the following observations: 1) Removing any loss degrades forecasting performance on all three datasets, indicating that the three losses complement each other and perform better when used jointly. 2) Replacing causal state mappings with pointwise MLPs consistently increases forecasting errors, suggesting that aggregating local temporal information through overlapping causal patches helps learn effective predicted states. 3) One-stage training using only observation-space MAE underperforms relative to the full model, suggesting that explicit latent-space supervision helps improve forecasting accuracy. 4) Removing covariates increases errors across datasets, especially on MMSP, highlighting the benefits of using external information.

6 Conclusion

Existing covariate-aware forecasting methods mainly use external information for observation-space forecasting, while the role of multimodal covariates in future-state prediction remains underexplored. Inspired by latent-space prediction in world models, WorldTS uses direct future-state supervision and two-stage training to learn latent space dynamics conditioned on multimodal covariates and to decode predicted future states into future observations. Experiments on 21 real-world datasets offer insight into the properties and the effectiveness of WorldTS; further analyses suggest that using covariates improves both forecasting accuracy and the quality of predicted future states.

AI USE STATEMENT

We used ChatGPT to help polish the manuscript and improve the clarity of the writing. All AI-assisted content was carefully reviewed by the authors. All methodological design and experimental analyses were conducted independently by the authors, and the use of AI was limited to auxiliary functions. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work focuses on methodological research for multimodal covariate-aware time series forecasting and does not involve human subjects or the collection of personal data. All experiments are conducted on existing benchmark datasets following established evaluation protocols. We do not release any new dataset containing sensitive or personally identifiable information. The proposed method is intended for general-purpose forecasting research, and we are not aware of any direct ethical risks arising from the methodology itself.

REPRODUCIBILITY STATEMENT

The WorldTS architecture and training procedure are described in Section 4, and the experimental settings are provided in Section 5.1. Dataset descriptions and implementation details are provided in Appendix A. Code for the main experiments is available at `https://anonymous.4open.science/r/WorldTS-C25A/`.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Türkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda-Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Trans. Mach. Learn. Res.*, 2024, 2024.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, pp. 15619–15629, 2023.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew J. Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *CoRR*, abs/2506.09985, 2025.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

Hanyin Cheng, Jingrong Zhou, Yang Shu, and Chenjuan Guo. KITE: Knowledge-guided probabilistic modeling for time series forecasting with exogenous variables. In *ICML*, 2026.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with TiDE: Time-series dense encoder. *Trans. Mach. Learn. Res.*, 2023, 2023.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *ICML*, volume 235, pp. 10148–10167, 2024.

Vijay Ekambaram, Arindam Jati, Nam Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. TSMixer: Lightweight mlp-mixer model for multivariate time series forecasting. In *SIGKDD*, pp. 459–469, 2023.

Vijay Ekambaram, Arindam Jati, Pankaj Dayama, Sumanta Mukherjee, Nam H Nguyen, Wesley M Gifford, Chandra Reddy, and Jayant Kalagnanam. Tiny time mixers (TTMs): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series. *NeurIPS*, 37:74147–74181, 2024.

Jingru Fei, Kun Yi, Wei Fan, Qi Zhang, and Zhendong Niu. Amplifier: Bringing attention to neglected low-energy components in time series forecasting. In *AAAI*, volume 39, pp. 11645–11653, 2025.

Raj Ghugare, Homanga Bharadhwaj, Benjamin Eysenbach, Sergey Levine, and Russ Salakhutdinov. Simplifying model-based RL: learning representations, latent-space models, and policies with one objective. In *ICLR*, 2023.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Lu Han, Yu Liu, Lan Li, Qiwen Deng, Jian Jiang, Yinbo Sun, Zhe Yu, Binfeng Wang, Xingyu Lu, Lintao Ma, Han-Jia Ye, et al. UniCA: Unified covariate adaptation for time series foundation model. In *ICLR*, volume 2026, pp. 6403–6443, 2026.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan Ö. Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. In *ICML*, volume 267, 2025.

Shi Bin Hoo, Samuel Müller, David Salinas, and Frank Hutter. The tabular foundation model TabPFN outperforms specialized time series forecasting models based on simple features. In *NeurIPS Workshop*, 2024.

Yifan Hu, Jie Yang, Tian Zhou, Peiyuan Liu, Yujin Tang, Rong Jin, and Liang Sun. Bridging past and future: Distribution-aware alignment for time series forecasting. In *ICLR*, volume 2026, pp. 126148–126172, 2026.

Songtao Huang, Zhen Zhao, Can Li, and Lei Bai. TimeKAN: Kan-based frequency decomposition learning architecture for long-term time series forecasting. In *ICLR*, volume 2025, pp. 28514–28529, 2025.

Furong Jia, Kevin Wang, Yixiang Zheng, Defu Cao, and Yan Liu. GPT4MTS: Prompt-based large language model for multimodal time-series forecasting. In *AAAI*, volume 38, pp. 23343–23351, 2024.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H. Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Afroz Mohiuddin, Ryan Sepassi, George Tucker, and Henryk Michalewski. Model based reinforcement learning for atari. In *ICLR*, 2020.

Seunghan Lee, Jun Seo, Jaehoon Lee, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, SoonYoung Lee, and Wonbin Ahn. Rethinking multimodal fusion for time series: Text modalities need constrained fusion. *arXiv preprint arXiv:2603.22372*, 2026.

Zhengyu Li, Xiangfei Qiu, Yuhan Zhu, Xingjian Wu, Jilin Hu, Chenjuan Guo, and Bin Yang. GCGNet: Graph-consistent generative network for time series forecasting with exogenous variables. In *ICLR*, 2026a.

Zihao Li, Xiao Lin, Zhining Liu, Jiaru Zou, Ziwei Wu, Lecheng Zheng, Dongqi Fu, Yada Zhu, Hendrik Hamann, Hanghang Tong, et al. Language in the flow of time: Time-series-paired texts weaved into a unified temporal narrative. In *ICLR*, volume 2026, pp. 24437–24484, 2026b.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*, 37(4): 1748–1764, 2021.

Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, et al. Time-MMD: Multi-domain multimodal dataset for time series analysis. In *NeurIPS*, volume 37, pp. 77888–77933, 2024a.

Peiyuan Liu, Hang Guo, Tao Dai, Naiqi Li, Jigang Bao, Xudong Ren, Yong Jiang, and Shu-Tao Xia. CALF: Aligning llms for time series forecasting via cross-modal fine-tuning. In *AAAI*, volume 39, pp. 18915–18923, 2025.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *ICLR*, volume 2024, pp. 11116–11140, 2024b.

Ziqing Ma, Wenwei Wang, Tian Zhou, Chao Chen, Bingqing Peng, Liang Sun, and Rong Jin. FusionSF: Fuse heterogeneous modalities in a vector quantized framework for robust solar power forecasting. In *SIGKDD*, pp. 5532–5543, 2024.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels. *CoRR*, abs/2603.19312, 2026.

Francisco Martínez-Álvarez, Alicia Troncoso Lora, José C. Riquelme, and Jesús S. Aguilar-Ruiz. Energy time series forecasting based on pattern sequence similarity. *IEEE Transactions on Knowledge and Data Engineering*, 23(8):1230–1243, 2011.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *ICLR*, 2023.

Kin G Olivares, Cristian Challu, Grzegorz Marcjasz, Rafał Weron, and Artur Dubrawski. Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx. *International Journal of Forecasting*, 39(2):884–900, 2023.

Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. TFB: towards comprehensive and fair benchmarking of time series forecasting methods. *Proc. VLDB Endow.*, 17(9):2363–2377, 2024.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual clustering enhanced multivariate time series forecasting. In *SIGKDD*, pp. 1185–1196, 2025.

Xiangfei Qiu, Yuhan Zhu, Zhengyu Li, Xingjian Wu, Bin Yang, and Jilin Hu. DAG: A dual correlation network for time series forecasting with exogenous variables. In *ICML*, 2026.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3): 1181–1191, 2020.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-scale time series foundation models with mixture of experts. In *ICLR*, volume 2025, pp. 34635–34667, 2025.

Siyuan Wang, Peng Chen, Yihang Wang, Wanghui Qiu, Chenjuan Guo, Bin Yang, and Yang Shu. Unlocking the value of text: Event-driven reasoning and multi-level alignment for time series forecasting. In *ICLR*, volume 2026, pp. 31182–31210, 2026.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. TimeXer: Empowering transformers for time series forecasting with exogenous variables. In *NeurIPS*, volume 37, pp. 469–498, 2024.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *ICML*, volume 235, pp. 53140–53164, 2024.

Binqing Wu, Weiqi Chen, Wengwei Wang, Bingqing Peng, Liang Sun, and Ling Chen. WeatherGNN: Exploiting meteo- and spatial-dependencies for local numerical weather prediction bias-correction. In *IJCAI*, pp. 2433–2441, 2024.

Ronghui Xu, Jihao Chen, Jindong Tian, Chenjuan Guo, and Bin Yang. MoST: A foundation model for multi-modality spatio-temporal traffic prediction. In *SIGKDD*, pp. 1716–1727, 2026.

Jie Yang, Yifan Hu, Yuante Li, Kexin Zhang, Kaize Ding, and Philip S Yu. From observations to states: Latent time series forecasting. *arXiv preprint arXiv:2602.00297*, 2026.

Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. Forecasting fine-grained air quality based on big data. In *SIGKDD*, pp. 2267–2276, 2015.

Siru Zhong, Weilin Ruan, Ming Jin, Huan Li, Qingsong Wen, and Yuxuan Liang. Time-VLM: Exploring multimodal vision-language models for augmented time series forecasting. In *ICML*, volume 267, 2025.

Pengfei Zhou, Yunlong Liu, Junli Liang, Qi Song, and Xiangyang Li. CrossLinear: Plug-and-play cross-correlation embedding for time series forecasting with exogenous variables. In *SIGKDD*, pp. 4120–4131, 2025.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In *NeurIPS*, 2023.

A Experimental Details

A.1 Datasets

Numerical-covariate datasets. For numerical-covariate forecasting, we use the 12 TSF-X datasets adopted by DAG (Qiu et al., 2026). Five of them are electricity-market benchmarks. NP uses the Nord Pool electricity price as the forecasting target and provides grid-load and wind-power forecasts as external variables. PJM focuses on the zonal price in the Commonwealth Edison area, accompanied by system-level and regional load forecasts. BE, FR, and DE contain electricity prices from the Belgian, French, and German markets, respectively; their covariates comprise market-specific load and generation forecasts, including wind and solar generation where available. The Energy dataset describes the hourly generation mix of the Chilean power system. We forecast thermoelectric output using generation from the other energy sources as covariates. Colbun and Rapel are daily hydropower datasets from two Chilean reservoirs, where the future water level is predicted from precipitation and tributary-inflow measurements. Finally, Sdwpfm1, Sdwpfm2, Sdwpfh1, and Sdwpfh2 are constructed from two wind turbines at two temporal resolutions. Their target is active power, and the external variables are ERA5 meteorological measurements, including temperature, surface pressure, relative humidity, wind conditions, and precipitation.

Text-covariate datasets. For text-covariate forecasting, we use eight domains from Time-MMD (Liu et al., 2024a): Agriculture, Climate, Economy, Energy, Health, Security, Social Good, and Traffic. Each domain associates a numerical target series with time-relevant reports or other textual records. Agriculture pairs broiler-market prices with USDA market reports; Climate combines NOAA drought indices with monthly climate reports; and Economy links international trade measurements with releases on trade and economic indicators. The Energy domain augments gasoline-price observations with reports published by the U.S. Energy Information Administration, while Health combines influenza-like-illness statistics with influenza surveillance material. Security concerns disaster and emergency records and is supplemented with disaster-related reports. Social Good uses U.S. unemployment statistics together with employment and labor-force publications, and Traffic pairs traffic-volume measurements with corresponding transportation reports. These datasets evaluate whether temporally associated language can provide useful context for predicting the evolution of a numerical target.

Image-covariate dataset. For image-covariate forecasting, we use the Multimodal Solar Power (MMSP) dataset (Ma et al., 2024), which contains approximately one and a half years of hourly observations from 88 photovoltaic plants. Following FusionSF (Ma et al., 2024) and UniCA (Han et al., 2026), we evaluate the first 10 plants. The target is future solar-power output, and the auxiliary information includes aligned satellite imagery, numerical weather predictions, and plant coordinates. The satellite observations provide spatial weather information beyond the historical power sequence, making MMSP a natural benchmark for visual and numerical covariates.

A.2 Baselines

To ensure a comprehensive and representative comparison, we compare WorldTS against a broad set of competitive baselines across numerical, text, and image covariates. These baselines include dedicated covariate-aware models, strong general-purpose forecasters, multimodal forecasting methods, and pretrained or foundation models, covering the major forecasting paradigms.

Numerical-covariate baselines. We compare WorldTS with methods covering several approaches to numerical covariate modeling. KITE (Cheng et al., 2026) uses external knowledge to guide probabilistic forecasting, whereas DAG (Qiu et al., 2026) explicitly models the relations between endogenous and exogenous series and between historical and forecast-period variables. GCGNet (Li et al., 2026a) represents cross-series dependencies through a graph-based generative model, and TimeXer (Wang et al., 2024) introduces variate-level representations through which a Transformer can incorporate exogenous series. TFT (Lim et al., 2021) combines variable selection, gating, and temporal attention to process observed and known-future inputs, while TiDE (Das et al., 2023) uses a dense encoder–decoder architecture with explicit access to dynamic covariates. To provide comparisons with general-purpose forecasting approaches, we include DUET (Qiu et al., 2025), which

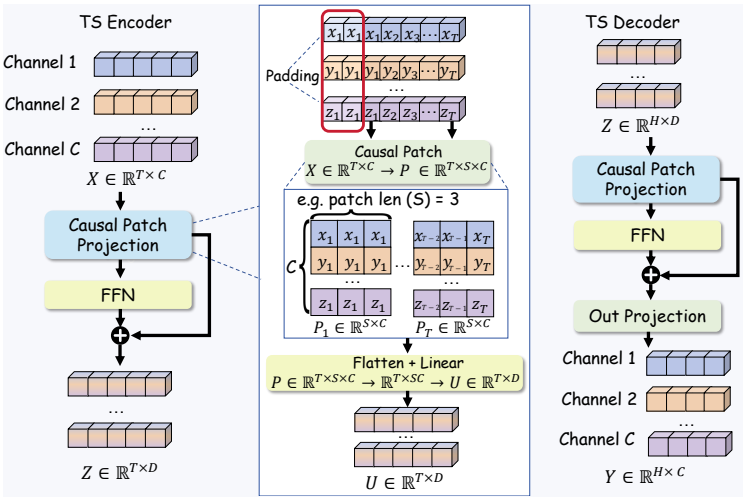

Figure 5: TS Encoder and Decoder. Overlapping causal observation patches yield time-aligned states, while local state patches reconstruct target observations.

learns relationships among dynamically clustered channels; CrossLinear (Zhou et al., 2025), which constructs cross-correlation representations across variables; Amplifier (Fei et al., 2025), which emphasizes informative low-energy signal components; and TimeKAN (Huang et al., 2025), which models temporal patterns after frequency decomposition. Together, these baselines cover both dedicated exogenous-variable models and general-purpose forecasting methods.

Text-covariate baselines. We first consider three methods that forecast from numerical histories without using the paired reports. RaFT (Han et al., 2025) augments prediction by retrieving relevant training sequences, iTransformer (Liu et al., 2024b) treats variables as tokens, and PatchTST (Nie et al., 2023) models the input through channel-independent temporal patches. We also compare with a range of multimodal time-series forecasting methods. VoT (Wang et al., 2026) extracts event-level evidence from text and aligns its textual and numerical predictions at multiple levels, whereas TaTS (Li et al., 2026b) organizes time-series-paired documents into a temporally coherent textual representation. GPT4TS (Zhou et al., 2023) adapts a pretrained language model to numerical time-series patterns, and GPT4MTS (Jia et al., 2024) formulates multimodal forecasting through prompt-based interaction with a large language model. Time-VLM (Zhong et al., 2025) uses a vision–language model to augment temporal prediction, while CALF (Liu et al., 2025) learns cross-modal alignment through targeted fine-tuning of a pretrained language model.

Image-covariate baselines. Following the MMSP comparison protocol of UniCA (Han et al., 2026), we include four families of methods. The specialized forecasters comprise PatchTST (Nie et al., 2023), N-BEATSx (Olivares et al., 2023), DeepAR (Salinas et al., 2020), TFT (Lim et al., 2021), and TiDE (Das et al., 2023). They span target-only patch modeling, autoregressive probabilistic prediction, and architectures that can consume structured historical or future covariates. TTM-R2 (Ekambaram et al., 2024), Moirai (Woo et al., 2024), and TabPFN-TS (Hoo et al., 2024) represent pretrained forecasting models with different MLP-mixing, Transformer, and tabular pretraining strategies. FusionSF (Ma et al., 2024) and MM-TSF (Liu et al., 2024a) are included as explicit multimodal forecasters. Finally, we report Chronos-Bolt (Ansari et al., 2024), TimesFM (Das et al., 2024), and Time-MoE (Shi et al., 2025) equipped with UniCA, which supplies a common covariate-adaptation interface to otherwise heterogeneous time-series foundation models. This collection contrasts WorldTS with task-specific models, pretrained forecasters, multimodal forecasting methods, and covariate-adapted foundation models under the same evaluation setting.

A.3 Implementation Details

Data splits and environment. Following DAG (Qiu et al., 2026), the numerical-covariate datasets are chronologically split into 70% training, 10% validation, and 20% test observations. Follow-

Algorithm 1 Pointwise Encoder–Decoder Ablation

Input: Training set $\mathcal{D}$; historical and future observations $x_{1:T}$ and $x_{T+1:T+H}$; multimodal covariates $\mathcal{C}$; pointwise MLP encoders $\bar{E}_\theta$ and $\bar{E}^{\mathrm{ts}}_{\theta_{\mathrm{ts}}}$; unchanged auxiliary encoders $\{E^m_{\theta_m}\}_{m\in\{\mathrm{txt},\mathrm{img}\}}$; state predictor P_ψ; pointwise MLP decoder $\bar{D}_\phi$; learning rates η_1, η_2; epochs N_1, N_2
Output: Forecast $\widehat{x}_{T+1:T+H}$
Stage 1: Latent Dynamics Learning
for $i = 1$ to N_1 **do**
  for $(x_{1:T}, x_{T+1:T+H}, \mathcal{C}) \in \mathcal{D}$ **do**
    $Z_{1:T} \leftarrow \bar{E}_\theta(x_{1:T})$
    $Z_{T+1:T+H} \leftarrow \bar{E}_\theta(x_{T+1:T+H})$
    Encode $\{Z^m\}_{m\in\mathcal{M}}$ from $\mathcal{C}$, using $\bar{E}^{\mathrm{ts}}_{\theta_{\mathrm{ts}}}$ for numerical covariates
    $\widehat{Z}_{T+1:T+H} \leftarrow P_\psi(Z_{1:T}, \{Z^m\}_{m\in\mathcal{M}})$
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{Stage1}}$
    Update the encoders and P_ψ using $\eta_1 \nabla\mathcal{L}$
  end for
end for
Freeze the encoders and P_ψ
Stage 2: Observation Decoding
for $i = 1$ to N_2 **do**
  for $(x_{1:T}, x_{T+1:T+H}, \mathcal{C}) \in \mathcal{D}$ **do**
    Compute $\widehat{Z}_{T+1:T+H}$ and $Z_{T+1:T+H}$ with the frozen modules
    $\widehat{x}_{T+1:T+H} \leftarrow \bar{D}_\phi(\widehat{Z}_{T+1:T+H})$
    $\widetilde{x}_{T+1:T+H} \leftarrow \bar{D}_\phi(Z_{T+1:T+H})$
    $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{Stage2}}$
    Update $\bar{D}_\phi$ using $\eta_2 \nabla\mathcal{L}$
  end for
end for
Inference: compute $\widehat{Z}_{T+1:T+H}$ from $x_{1:T}$ and $\mathcal{C}$
return $\widehat{x}_{T+1:T+H} = \bar{D}_\phi(\widehat{Z}_{T+1:T+H})$

ing VoT (Wang et al., 2026), Time-MMD adopts the same 70%/10%/20% chronological training/validation/test split. Following UniCA (Han et al., 2026), MMSP is chronologically split into 80% training, 10% validation, and 10% test observations. All experiments are implemented in PyTorch and run on NVIDIA A800 80GB GPUs. The numerical-covariate experiments use Python 3.8, while the text- and image-covariate experiments use Python 3.10. Both stages use Adam, with seed 2021 for numerical-covariate experiments and 2025 for Time-MMD and MMSP.

Implementation details. Following DAG (Qiu et al., 2026), we evaluate numerical-covariate forecasting under both short-term and long-term settings. For Colbun and Rapel, short-term forecasting uses a lookback window of 60 steps with a prediction horizon of 10 steps, while long-term forecasting uses a lookback window of 180 steps with a prediction horizon of 30 steps. For all other datasets, short-term forecasting uses a lookback window of 168 steps with a prediction horizon of 24 steps, while long-term forecasting uses a lookback window of 720 steps with a prediction horizon of 360 steps. Following VoT (Wang et al., 2026), we use a lookback window of 36 steps for Energy and Health and 8 steps for the remaining six Time-MMD datasets. We select a prediction horizon of 12 steps for all eight datasets. Following UniCA (Han et al., 2026), we evaluate image-covariate forecasting on MMSP using a 2,048-step lookback window and a 24-step prediction horizon.

Modality encoders. The text and image encoders in Eq. 3 are instantiated as:

$$\begin{aligned} Z^{\mathrm{txt}}_{1:T} &= E^{\mathrm{txt}}_{\theta_{\mathrm{txt}}}\left(c^{\mathrm{txt}}_{1:T}\right) = \mathrm{MLP}_{\mathrm{txt}}\big(\mathrm{MeanPool}\big(\mathrm{Embed}_{\mathrm{GPT2}}\big(c^{\mathrm{txt}}_{1:T}\big)\big)\big)\,, \\ Z^{\mathrm{img}}_{1:T} &= E^{\mathrm{img}}_{\theta_{\mathrm{img}}}\left(c^{\mathrm{img}}_{1:T}\right) = \mathrm{MLP}_{\mathrm{img}}\Big(\mathrm{Linear}_{\mathrm{img}}\Big(\mathrm{CNN}\Big(c^{\mathrm{img}}_{1:T}\Big)\Big)\Big) \end{aligned} \tag{22}$$

Following VoT (Wang et al., 2026), the text branch uses the input embedding layer of a frozen GPT-2 model to obtain token-level representations and mean pools the valid tokens at each time step. The

Algorithm 2 One-Stage Observation-Space MAE Training

Input: Training set $\mathcal{D}$; historical and future observations $x_{1:T}$ and $x_{T+1:T+H}$; multimodal covariates $\mathcal{C}$; state encoder E_θ; modality encoders $\{E^m_{\theta_m}\}_{m\in\mathcal{M}}$; state predictor P_ψ; state decoder D_ϕ; learning rate η; epochs $N_1 + N_2$
Output: Forecast $\widehat{x}_{T+1:T+H}$
End-to-end training:
for $i = 1$ to $N_1 + N_2$ **do**
  for $(x_{1:T}, x_{T+1:T+H}, \mathcal{C}) \in \mathcal{D}$ **do**
    $Z_{1:T} \leftarrow E_\theta(x_{1:T})$
    $\{Z^m\}_{m\in\mathcal{M}} \leftarrow \{E^m_{\theta_m}(c^m)\}_{m\in\mathcal{M}}$
    $\widehat{Z}_{T+1:T+H} \leftarrow P_\psi(Z_{1:T}, \{Z^m\}_{m\in\mathcal{M}})$
    $\widehat{x}_{T+1:T+H} \leftarrow D_\phi(\widehat{Z}_{T+1:T+H})$
    $\mathcal{L}_{\text{obs}} \leftarrow \|\widehat{x}_{T+1:T+H} - x_{T+1:T+H}\|_1$
    Jointly update θ, $\{\theta_m\}$, ψ, and ϕ using $\eta\nabla\mathcal{L}_{\text{obs}}$
  end for
end for
Inference: compute $\widehat{x}_{T+1:T+H}$ from $x_{1:T}$ and $\mathcal{C}$
return $\widehat{x}_{T+1:T+H}$

pooled features are mapped by $\mathrm{MLP}_{\mathrm{txt}}$ to the time-aligned text representations $Z^{\mathrm{txt}}_{1:T}$. Following UniCA (Han et al., 2026), the image branch uses a lightweight CNN to extract frame-wise spatial features. $\mathrm{Linear}_{\mathrm{img}}$ converts the heterogeneous visual features into compact continuous covariate representations, which are mapped by $\mathrm{MLP}_{\mathrm{img}}$ to the time-aligned image representations $Z^{\mathrm{img}}_{1:T}$.

A.4 Ablation Protocols

For the ablation studies, we set the prediction horizon to 10 for Colbun, 12 for Climate, and 24 for MMSP. In Algorithms 1 and 2, c^m and Z^m denote $c^m_{1:\tau_m}$ and $Z^m_{1:\tau_m}$.

Pointwise encoder–decoder. We replace the state encoder, numerical-covariate encoder, and state decoder with two-layer pointwise MLPs. At each time step, the state encoder maps an observation $x_t \in \mathbb{R}^C$ to a state $Z_t \in \mathbb{R}^D$, while the decoder maps the predicted state back to the observation space:

$$Z_t = \bar{E}_\theta(x_t), \qquad \widehat{x}_t = \bar{D}_\phi\Big(\widehat{Z}_t\Big) \tag{23}$$

These mappings preserve the original input and output dimensions and share parameters across time, but do not aggregate information from neighboring time steps. The text and image encoders, state predictor, learning objectives, and two-stage training protocol remain unchanged. Algorithm 1 summarizes this variant.

One-stage observation-space training. To examine the effect of explicit predictive-state learning and the two-stage strategy, we remove the future-state encoding branch and its state-space objectives. The original causal patch encoder and decoder are retained, and the complete forecasting path is optimized end to end using only observation-space MAE. We train for $N_1 + N_2$ epochs to match the total number of training epochs of the full model. The resulting procedure is given in Algorithm 2.

For each loss ablation, only the corresponding coefficient λ_k, $k \in \{\mathrm{latent}, \mathrm{sim}, \mathrm{VIC}\}$, is set to zero. For the variant without covariates, we set $\mathcal{C} = \varnothing$ while retaining the full architecture, objectives, and two-stage training protocol.

B Full Numerical-Covariate Results

Table 5 reports the complete MSE and MAE results for all forecasting horizons on the 12 numerical-covariate datasets, together with the averages reported in the main text. WorldTS achieves the best or tied-best average performance on all datasets; for the few individual forecasting settings where it does not rank first, it typically remains second-best or within a competitive range.

Table 5: Full results on the 12 numerical-covariate datasets. *Avg* denotes the average result over the two forecasting horizons. Baseline results follow the same evaluation protocol as Table 1. The best results are in **red and bold**, and the second-best results are blue and underlined.

Models		**WorldTS** (ours)		KITE (2026)		DAG (2026)		GCGNet (2026)		DUET (2025)		CrossLinear (2025)		Amplifier (2025)		TimeKAN (2025)		TimeXer (2024)		TiDE (2023)		TFT (2021)	
Metrics		mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae
NP	24	**0.171**	**0.217**	0.179	0.223	0.202	0.237	0.208	0.237	0.246	0.287	0.210	0.266	0.252	0.303	0.273	0.310	0.236	0.266	0.284	0.301	0.219	0.249
	360	**0.423**	**0.420**	0.471	0.422	0.521	0.451	0.531	0.459	0.576	0.528	0.531	0.508	0.587	0.534	0.538	0.529	0.600	0.475	0.601	0.498	0.539	0.501
	Avg	**0.297**	**0.319**	0.325	0.323	0.362	0.344	0.370	0.348	0.411	0.407	0.371	0.387	0.420	0.418	0.405	0.419	0.418	0.371	0.443	0.400	0.379	0.375
PJM	24	**0.056**	**0.141**	**0.056**	0.143	0.057	0.143	0.060	0.150	0.072	0.166	0.088	0.191	0.096	0.208	0.115	0.244	0.075	0.166	0.106	0.214	0.095	0.195
	360	**0.118**	0.216	0.135	**0.214**	0.130	0.218	0.129	0.223	0.131	0.228	0.135	0.254	0.177	0.285	0.162	0.281	0.140	0.231	0.177	0.279	0.133	0.219
	Avg	**0.087**	**0.178**	0.096	**0.178**	0.093	0.180	0.095	0.187	0.102	0.197	0.112	0.223	0.137	0.246	0.139	0.262	0.108	0.198	0.142	0.246	0.114	0.207
BE	24	**0.340**	**0.226**	0.348	0.240	0.361	0.229	0.350	0.248	0.432	0.272	0.391	0.259	0.471	0.339	0.451	0.319	0.392	0.253	0.426	0.285	0.426	0.272
	360	**0.447**	**0.293**	0.507	0.332	0.485	0.330	0.511	0.340	0.597	0.436	0.568	0.416	0.646	0.487	0.645	0.495	0.512	0.327	0.571	0.364	0.482	0.310
	Avg	**0.393**	**0.260**	0.427	0.286	0.423	0.279	0.431	0.294	0.514	0.354	0.479	0.337	0.559	0.413	0.548	0.407	0.452	0.290	0.498	0.325	0.454	0.291
FR	24	0.337	0.176	**0.311**	0.178	0.355	**0.171**	0.347	0.188	0.384	0.251	0.390	0.226	0.459	0.348	0.454	0.296	0.366	0.208	0.418	0.255	0.543	0.253
	360	**0.431**	**0.244**	0.462	0.271	0.473	0.268	0.482	0.279	0.607	0.403	0.575	0.370	0.648	0.468	0.641	0.452	0.489	0.273	0.551	0.308	0.465	0.261
	Avg	**0.384**	**0.210**	0.387	0.225	0.414	0.219	0.415	0.234	0.495	0.327	0.482	0.298	0.554	0.408	0.547	0.374	0.427	0.241	0.484	0.281	0.504	0.257
DE	24	0.274	0.323	**0.262**	0.327	0.277	**0.322**	0.280	0.331	0.376	0.378	0.387	0.396	0.394	0.407	0.399	0.412	0.339	0.362	0.367	0.383	0.380	0.383
	360	**0.407**	**0.382**	0.437	0.413	0.462	0.418	0.523	0.447	0.589	0.482	0.583	0.507	0.551	0.474	0.547	0.479	0.610	0.474	0.630	0.511	0.599	0.509
	Avg	**0.340**	**0.353**	0.350	0.370	0.370	0.370	0.401	0.389	0.482	0.430	0.485	0.452	0.473	0.441	0.473	0.445	0.475	0.418	0.499	0.447	0.489	0.446
Energy	24	**0.039**	**0.151**	0.071	0.208	0.079	0.215	0.081	0.221	0.117	0.283	0.241	0.418	0.138	0.306	0.135	0.298	0.122	0.273	0.103	0.248	0.093	0.235
	360	**0.142**	**0.300**	0.151	0.305	0.169	0.320	0.182	0.334	0.288	0.452	0.237	0.385	0.328	0.472	0.302	0.464	0.204	0.357	0.202	0.355	0.167	0.331
	Avg	**0.091**	**0.226**	0.111	0.257	0.124	0.267	0.131	0.277	0.203	0.367	0.239	0.401	0.233	0.389	0.218	0.381	0.163	0.315	0.153	0.302	0.130	0.283
Sdwpfm1	24	0.384	**0.399**	0.384	0.410	**0.351**	0.400	0.376	0.415	0.551	0.564	0.355	0.473	0.364	0.445	0.418	0.503	0.558	0.533	0.474	0.488	0.366	0.421
	360	**0.398**	**0.425**	0.450	0.491	0.495	0.522	0.472	0.499	0.646	0.577	0.497	0.532	0.510	0.535	0.476	0.566	0.845	0.684	0.492	0.526	0.597	0.528
	Avg	**0.391**	**0.412**	0.417	0.451	0.423	0.461	0.424	0.457	0.599	0.570	0.426	0.502	0.437	0.490	0.447	0.534	0.701	0.609	0.483	0.507	0.482	0.474
Sdwpfm2	24	0.417	**0.414**	0.443	0.450	**0.372**	**0.414**	0.421	0.441	0.445	0.452	0.477	0.536	0.394	0.462	0.474	0.538	0.627	0.570	0.461	0.492	0.411	0.458
	360	**0.474**	**0.486**	0.506	0.515	0.583	0.556	0.529	0.531	0.584	0.528	0.589	0.611	0.587	0.563	0.520	0.589	0.978	0.736	0.511	0.540	0.541	0.519
	Avg	**0.446**	**0.450**	0.475	0.483	0.477	0.485	0.475	0.486	0.514	0.490	0.533	0.573	0.491	0.512	0.497	0.564	0.803	0.653	0.486	0.516	0.476	0.488
Sdwpfh1	24	**0.394**	**0.435**	0.424	0.464	0.408	0.438	0.435	0.486	0.527	0.513	0.548	0.585	0.576	0.627	0.511	0.582	0.651	0.587	0.434	0.489	0.401	0.460
	360	**0.357**	**0.428**	0.457	0.517	0.489	0.534	0.465	0.514	0.551	0.519	0.566	0.601	0.497	0.569	0.643	0.694	0.841	0.700	0.472	0.527	0.557	0.523
	Avg	**0.376**	**0.431**	0.441	0.491	0.448	0.486	0.450	0.500	0.539	0.516	0.557	0.593	0.537	0.598	0.577	0.638	0.746	0.643	0.453	0.508	0.479	0.491
Sdwpfh2	24	**0.406**	**0.451**	0.477	0.498	0.438	0.465	0.473	0.506	0.629	0.563	0.468	0.540	0.473	0.533	0.580	0.614	0.820	0.677	0.579	0.553	0.474	0.493
	360	**0.432**	**0.470**	0.524	0.555	0.608	0.595	0.566	0.565	0.665	0.569	0.608	0.609	0.569	0.628	0.713	0.729	0.962	0.761	0.619	0.614	0.657	0.549
	Avg	**0.419**	**0.461**	0.500	0.526	0.523	0.530	0.520	0.536	0.647	0.566	0.538	0.574	0.521	0.581	0.647	0.672	0.891	0.719	0.599	0.583	0.566	0.521
Colbun	10	**0.052**	**0.080**	0.055	0.101	0.061	0.094	0.065	0.108	0.089	0.134	0.071	0.102	0.071	0.121	0.061	0.101	0.113	0.172	0.089	0.131	0.092	0.135
	30	0.125	**0.213**	**0.121**	0.243	0.135	0.215	0.149	0.243	0.307	0.397	0.182	0.288	0.275	0.370	0.195	0.249	0.176	0.299	0.240	0.322	0.383	0.460
	Avg	**0.088**	**0.146**	**0.088**	0.172	0.098	0.154	0.107	0.175	0.198	0.266	0.126	0.195	0.173	0.246	0.128	0.175	0.145	0.235	0.164	0.227	0.238	0.297
Rapel	10	0.170	**0.200**	0.192	0.231	**0.151**	0.203	0.211	0.230	0.174	0.219	0.163	0.209	0.181	0.227	0.174	0.231	0.301	0.308	0.228	0.271	0.201	0.253
	30	**0.286**	**0.317**	0.296	0.358	0.309	0.408	0.401	0.384	0.365	0.432	0.340	0.417	0.333	0.416	0.325	0.390	0.387	0.416	0.411	0.432	0.409	0.414
	Avg	**0.228**	**0.259**	0.244	0.294	0.230	0.305	0.306	0.307	0.269	0.326	0.252	0.313	0.257	0.321	0.249	0.311	0.344	0.362	0.320	0.351	0.305	0.333

C Limitations and Future Work

Despite encouraging empirical results, a rigorous theoretical understanding of covariate-conditioned predictive-state learning remains open. Future work could formalize the links among multimodal covariate information, predictive-state quality, and forecasting error.

WorldTS is an initial attempt to adapt latent-space predictive learning from world models to time-series forecasting. Its objectives and regularization largely follow general-purpose designs; regularization tailored to temporal structure could better capture temporal dependencies and dynamics.

WorldTS currently targets regularly sampled series and is trained separately for each dataset. Integrating this approach with pretrained time-series foundation models and extending it to irregular sampling and asynchronous covariates could improve transferability and applicability.